Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский политехнический университет»

Кафедра «Инфокогнитивные технологии»

Образовательная программа «Веб-технологии»

Отчет по курсовому проекту

по дисциплине «Основы инженерного проектирования»

Тема: «Электронная регистратура Рязанской области»

**Выполнил:**

Студент группы 191-321

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Огородников Н. А.

подпись, дата

**Принял:**

Старший преподаватель

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Даньшина М.В.

подпись, дата

Москва 2021

Оглавление

[Оглавление 2](#_Toc62241169)

[Введение 3](#_Toc62241170)

[Тип разработки 3](#_Toc62241171)

[Задача 3](#_Toc62241172)

[Аналоги 3](#_Toc62241173)

[Основная часть 4](#_Toc62241174)

[Анализ аналогов 4](#_Toc62241175)

[Проектирование физической структуры 6](#_Toc62241176)

[Создание django-приложения 7](#_Toc62241177)

[Настройка административного интерфейса Django 9](#_Toc62241178)

[Наполнение базы данных. 13](#_Toc62241179)

[Реализация REST API 14](#_Toc62241180)

[Реализация импорта и экспорта данных 19](#_Toc62241181)

[Типовые запросы к базе данных 20](#_Toc62241182)

[Заключение 24](#_Toc62241183)

[Список используемых источников 25](#_Toc62241184)

Введение

Тип разработки

Индивидуальный проект, направленный на демонстрацию и фиксацию навыков, полученных в течение 3-го семестра обучения. Разработка ведется поэтапно.

Задача

Спроектировать и создать реляционную базу данных для веб-сервиса анкетирования пользователей сайта и настроить административную панель для управления базой данных с помощью фреймворка Django. Административная панель поможет создавать анкеты и вопросы, контролировать их содержание и просматривать ответы пользователей.

Аналоги

* [Google Forms](https://www.google.com/intl/ru_ua/forms/about/)
* [Survio](https://www.survio.com/ru/)
* [EXAMINARE](https://www.examinare.ru/)
* [Simpoll](https://simpoll.ru/)
* [Анкетолог](https://anketolog.ru/)

Основная часть

Анализ аналогов

Сервисы анкетирования широко распространены в наше время, поэтому у разрабатываемого решения есть множество аналогов. Вот некоторые из них:

* [Google Forms](https://www.google.com/intl/ru_ua/forms/about/)
* [Survio](https://www.survio.com/ru/)
* [EXAMINARE](https://www.examinare.ru/)
* [Simpoll](https://simpoll.ru/)
* [Анкетолог](https://anketolog.ru/)

Данные сервисы имеют схожую структуру, и процесс создания анкеты в них сводится к трем основным шагам:

1. Создание анкеты
2. Добавление секций анкеты (минимум 1)
3. Добавление вопросов (минимум 1)

Данные сервисы предоставляют широкие возможности по настройке анкет: добавление описаний, настройка типов вопросов, распределение на логические элементы (секции).

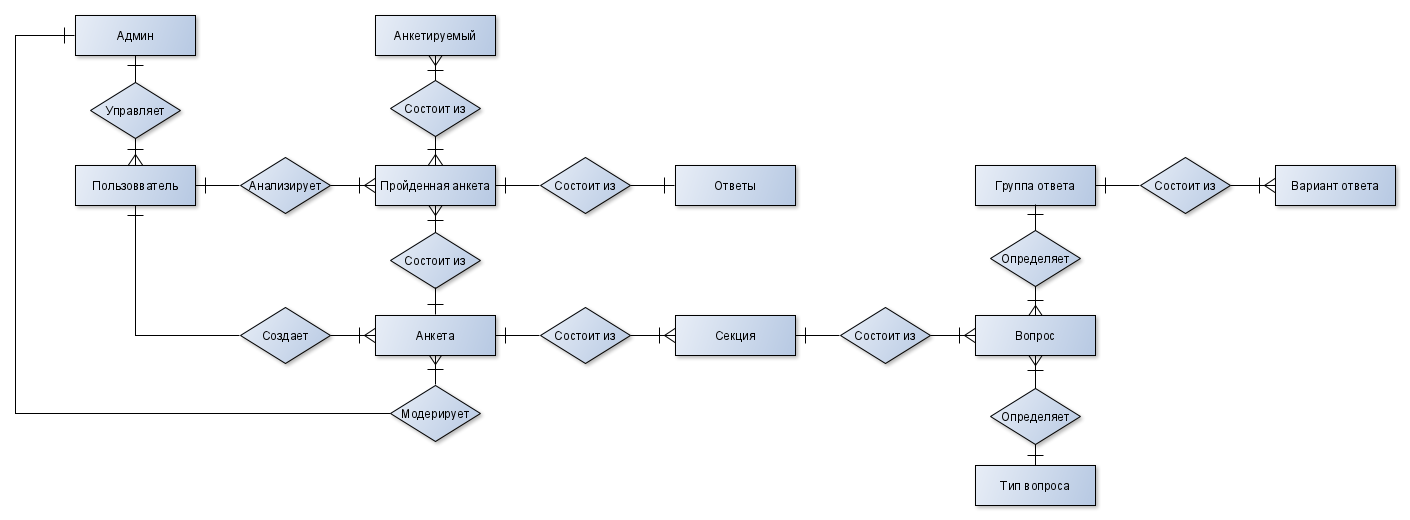
Все сервисы позволяют просматривать ответы, предоставленные пользователями, с целью дальнейшего анализа. Также есть поддержка экспорта ответов в большинстве современных форматов.

Опыт использования сервисов положительный. Интерфейс удобный и интуитивно понятный, присутствует адаптивность, что позволяет использовать сервисы на мобильных устройствах.

Подводя итог можно сказать, что сайты являются типовыми решениями, и имеют между собой высокую конкуренцию. Данные сервисы можно использовать как ориентир при разработке проекта.

**Проектирование инфологической модели предметной области**

Следуя из анализа конкурентов, сервис должен строится на двух основных процессах: создание анкет и анализ ответов. Анкета состоит из секций, которые состоят из вопросов. Вопрос может быть определенного типа (один из, несколько из, ответ в свободной форме и так далее), также вопрос принадлежит к какой-либо группе ответа (в нее входят варианты ответа на вопрос). Группа ответов снижает количество строк в базе данных при создании анкет с вопросами, предполагающими одинаковые варианты ответов («согласен/не согласен/не знаю» и подобные).

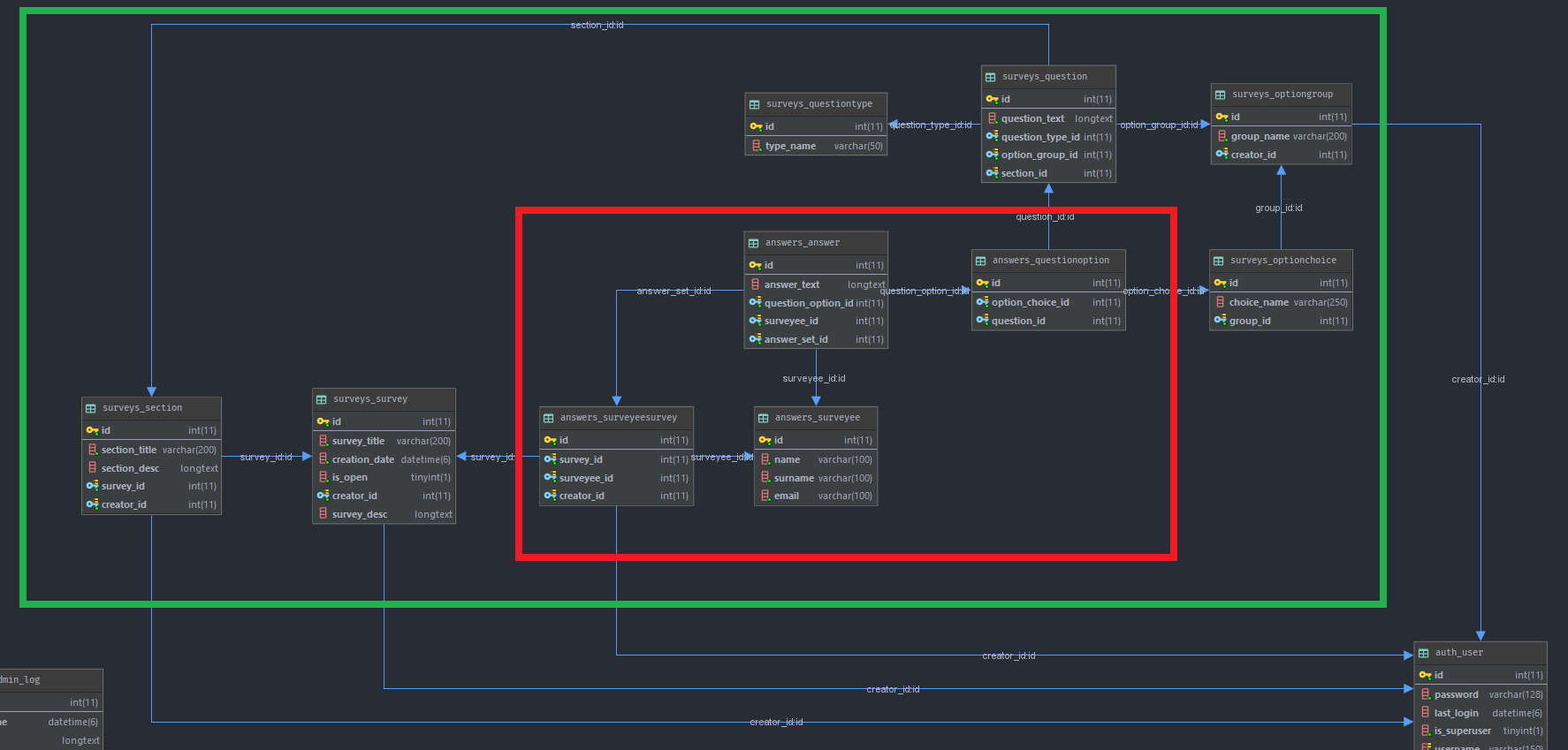


1. Инфологическая модель

Проектирование физической структуры

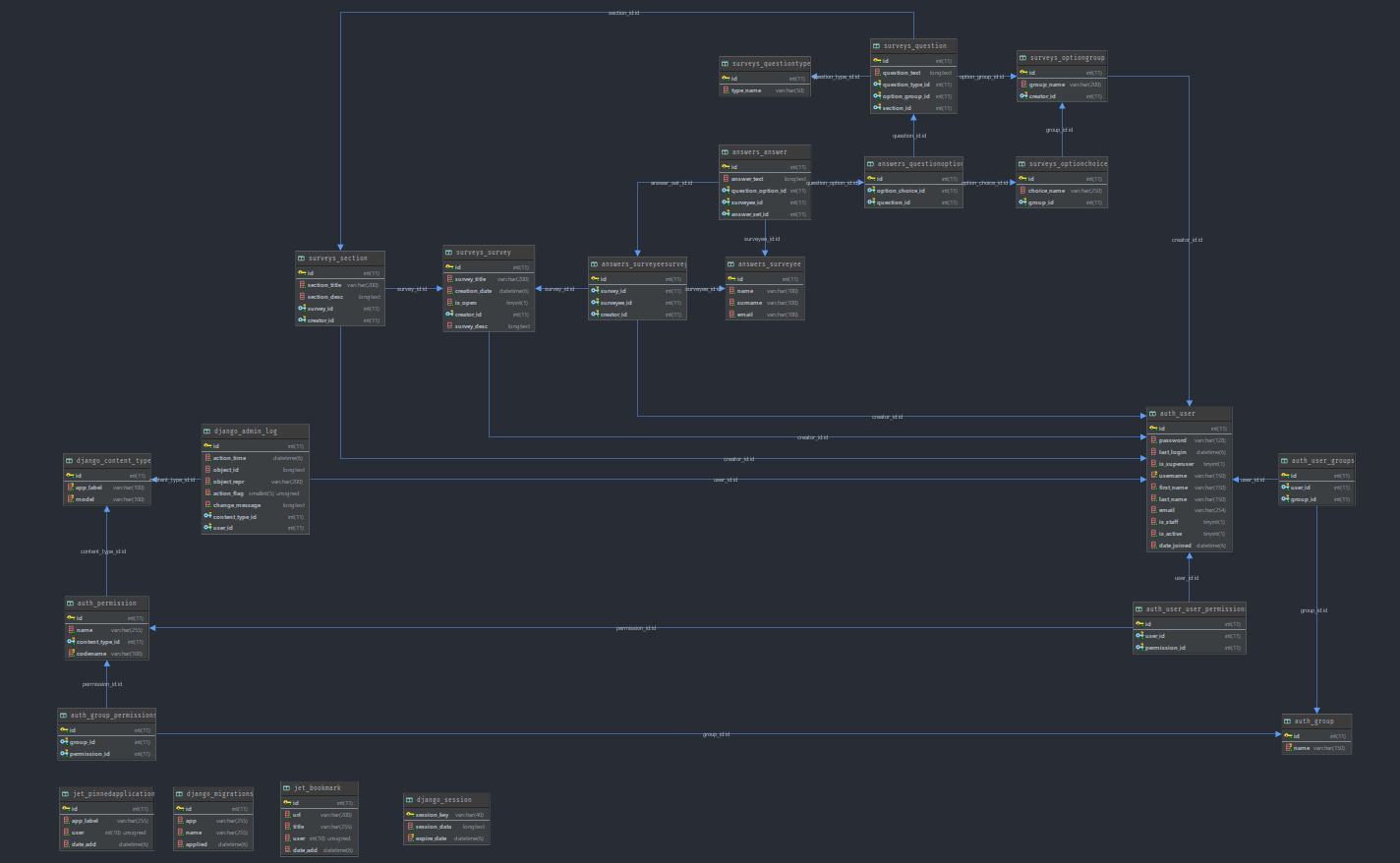
С помощью инфологической схемы можно составить подробную физическую схему базы данных. В ней находятся все те же сущности, но каждая из них имеет набор полей с определенным типом данных. Сущности связаны между собой по принципу, описанному в предыдущем пункте, однако имеется еще несколько связей с элементами базы данных фреймворка Django, обеспечивающие правильное функционирование системы.

Красным выделена часть базы данных, отвечающая за приложение ответов; зеленым – создания анкет.



2. Физическая структура

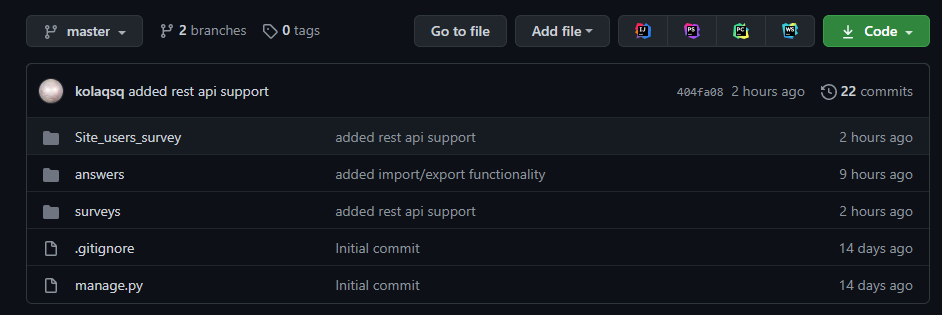
Полная схема базы данных, включающая таблицы фреймворка и его модулей выглядит следующим образом:



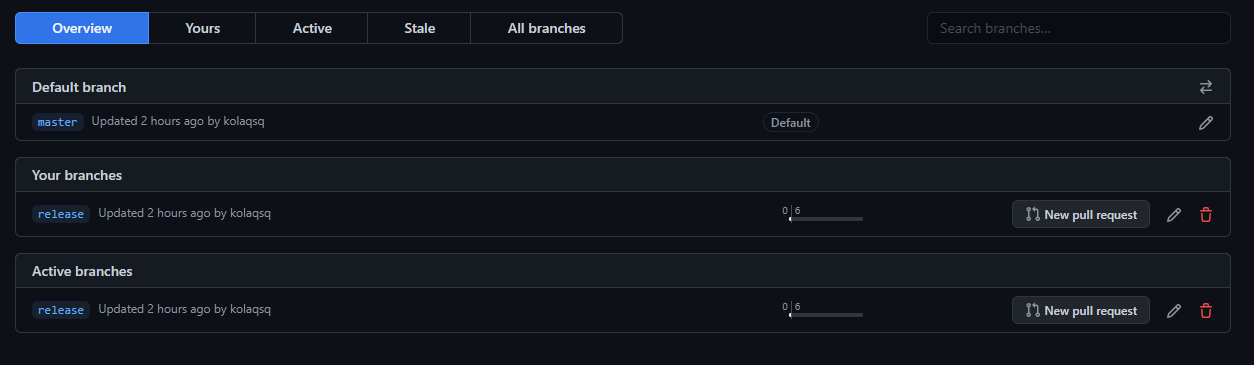
3. Полная схема БД

Создание django-приложения

После создания проект был загружен в GitHub в собственный репозиторий по адресу <https://github.com/kolaqsq/Site_users_survey>. В процессе разработки были была добавлена ветка release, в которую загружалась версия приложения, настроенная под использования на веб-сервере fit.mospolytech.ru.

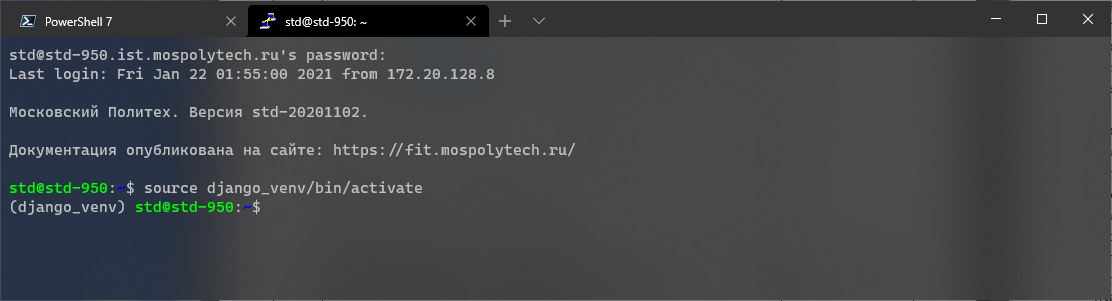


4. Git-репозиторий



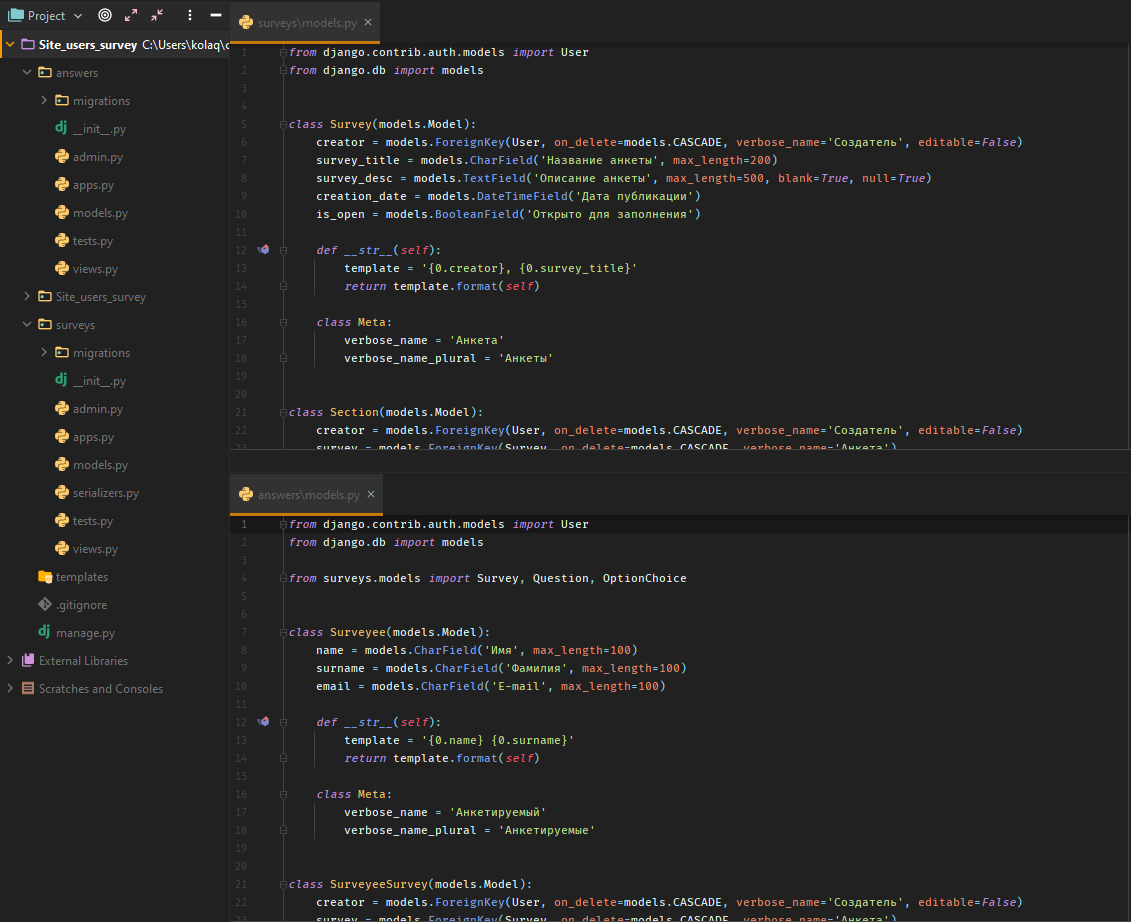
5. Ветки

На сервере было создано виртуальное python-окружение и установлены модули, необходимые для функционирования сервиса.



6. Виртуальное окружение

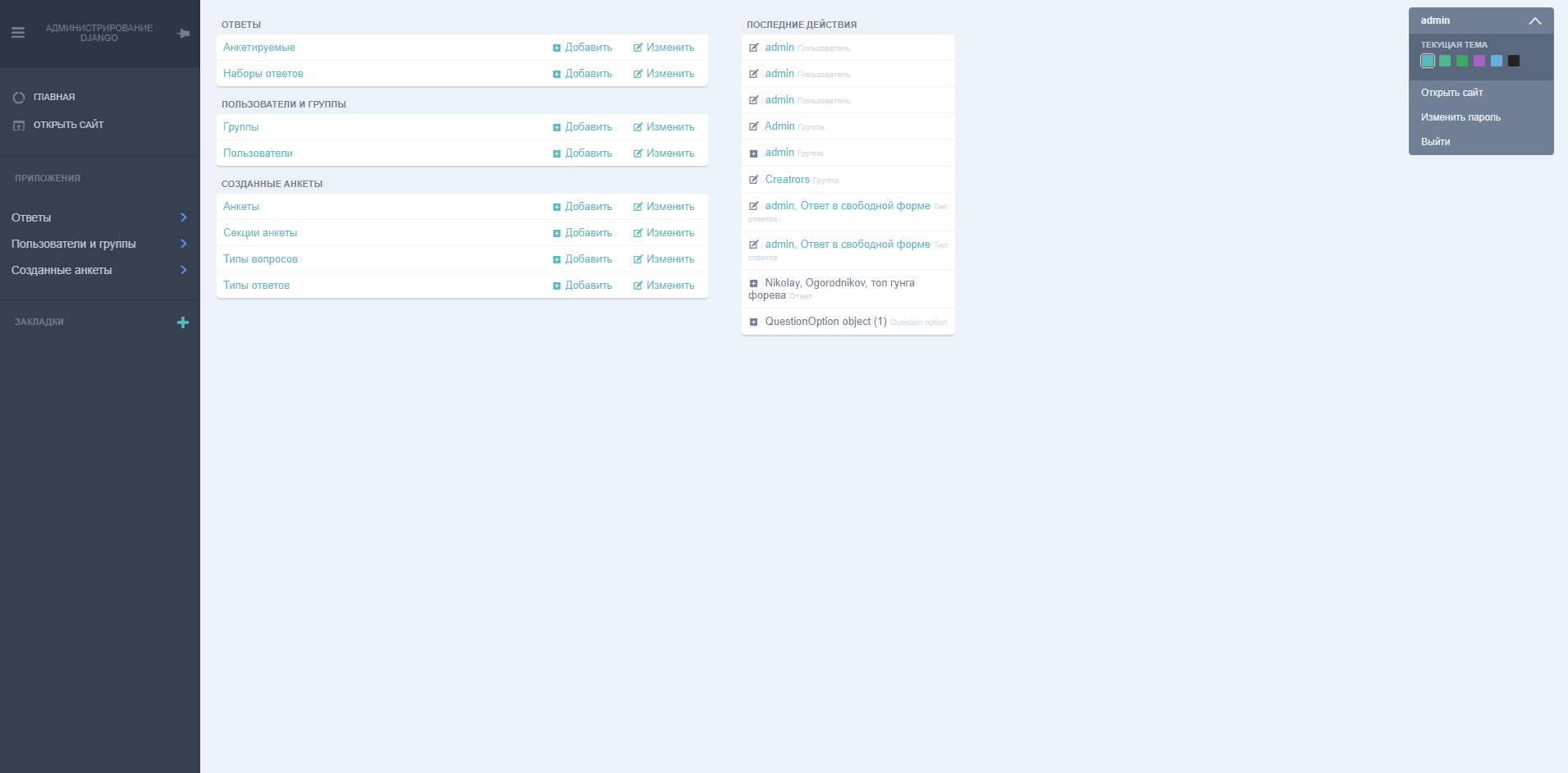
Далее в проекте было создано 2 приложения: surveys и answers. Первое нужно для создания и изменения анкет, а второе – для просмотра ответов анкетируемых. В процессе создания приложений в них были перенесены таблицы в качестве моделей.



7. Структура проекта и models.py

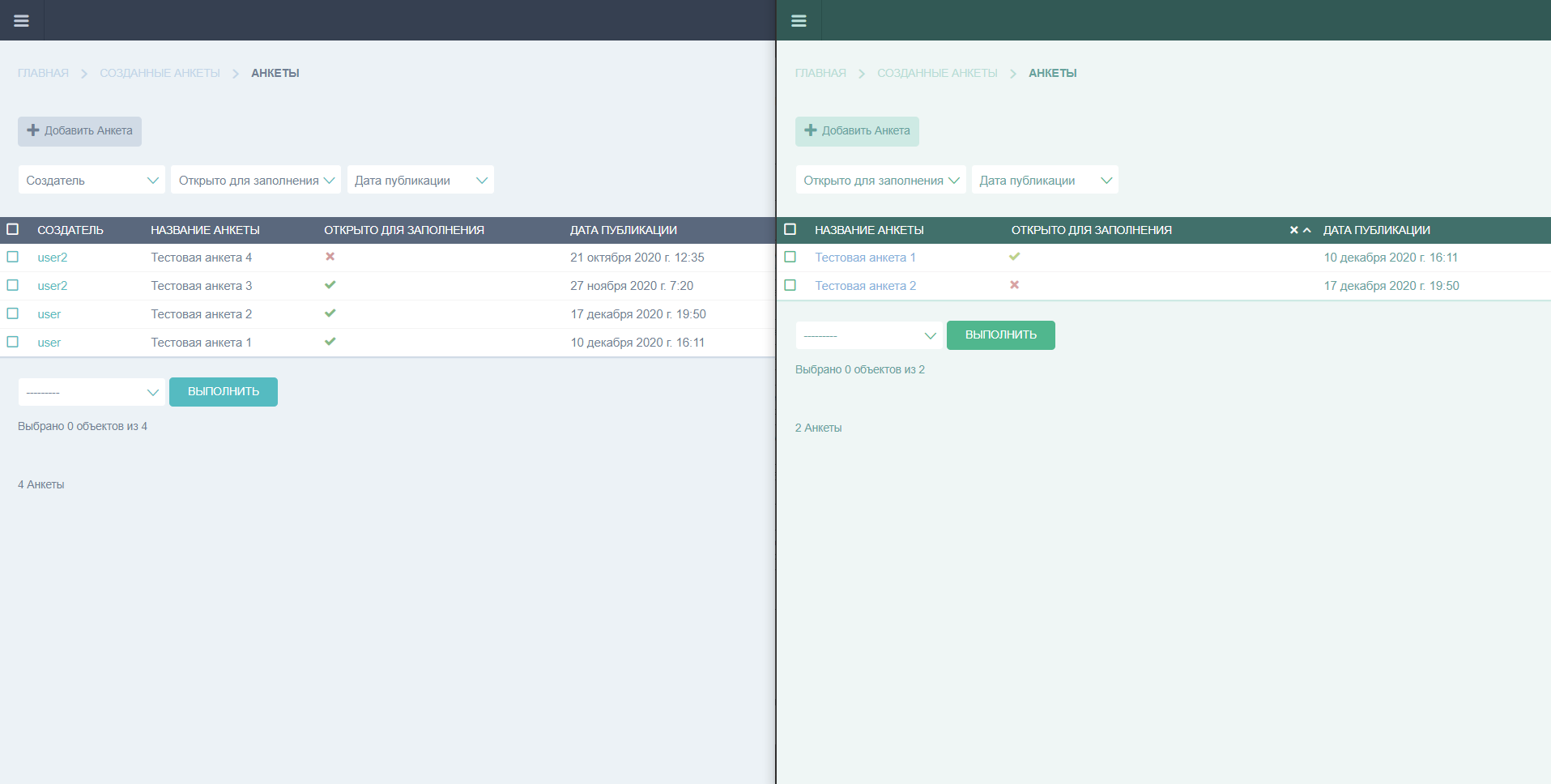
Настройка административного интерфейса Django

После создания приложения был установлен сторонний графический интерфейс Django JET, поскольку мне он показался более удобным.



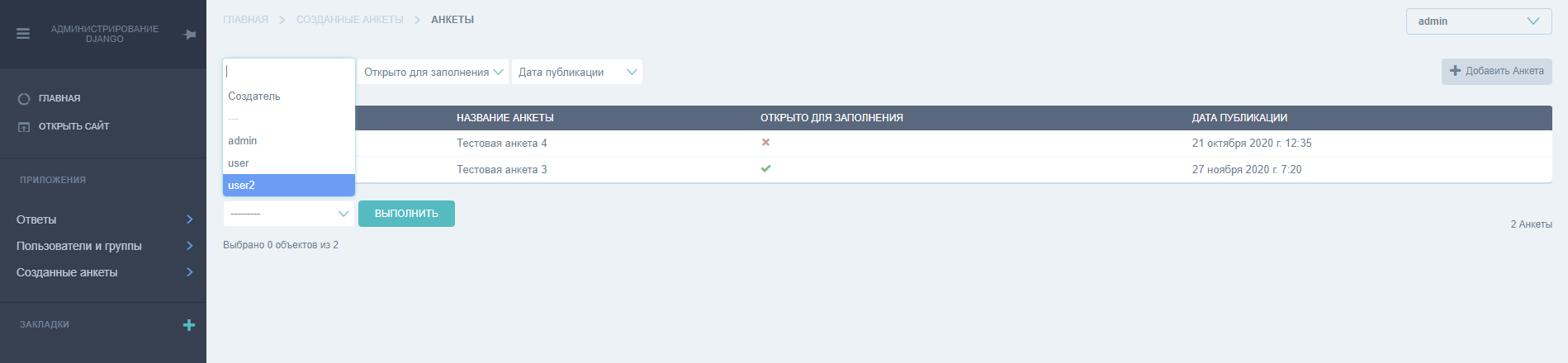
8. Интерфейс Django JET

Далее были созданы пользователи: admin/admin (суперпользователь), user/user и user2/user2 (обычные пользователи системы). Суперпользователь может видеть все сущности и редактировать их. Обычные пользователи могут создавать и редактировать сущности, принадлежащие этому пользователю, в приложении анкет. В приложении ответов обычный пользователь может только просматривать ответы на созданные им анкеты и список анкетируемых людей. Они также могут удалять ответы.

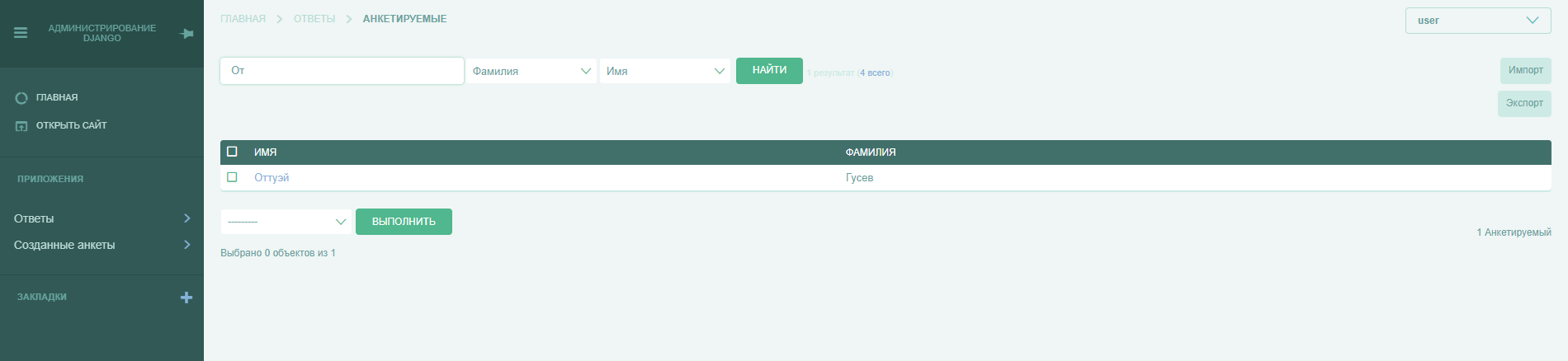


9. Отличия интерфейса у admin (слева) и user (справа)

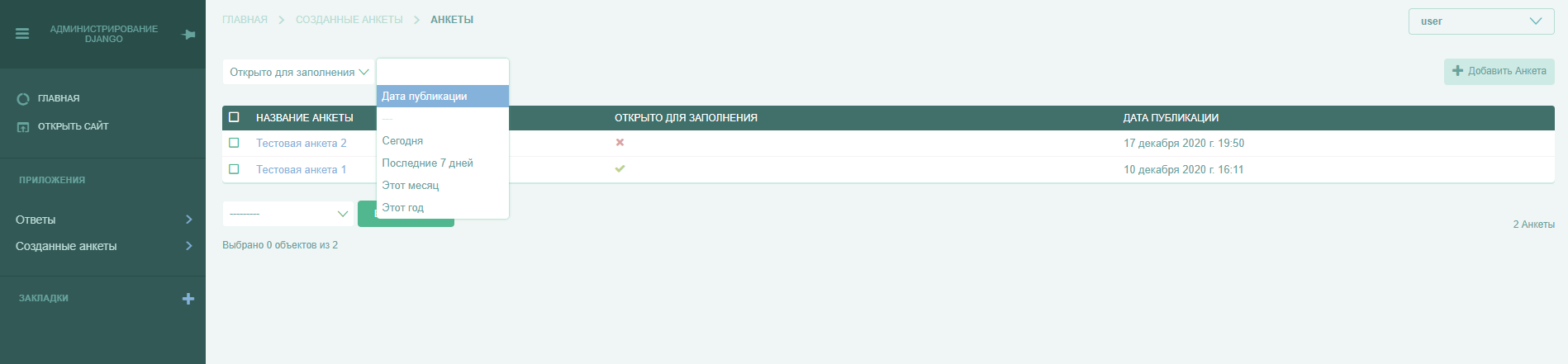
В процессе настройки интерфейса были применены такие методы как: фильтрация, поиск, фильтр по дате, сортировка, вывод информации из других таблиц, вложенность моделей.



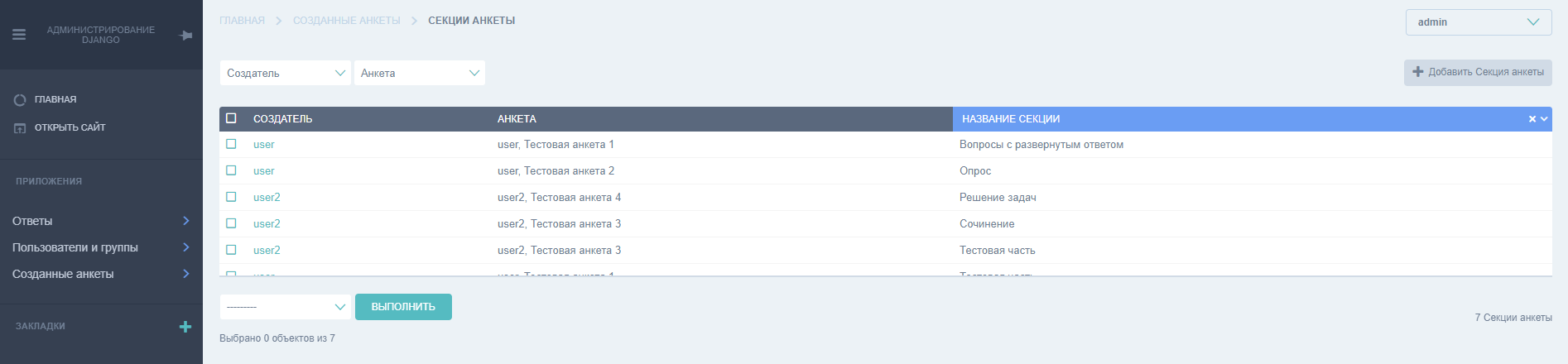
10. Фильтрация



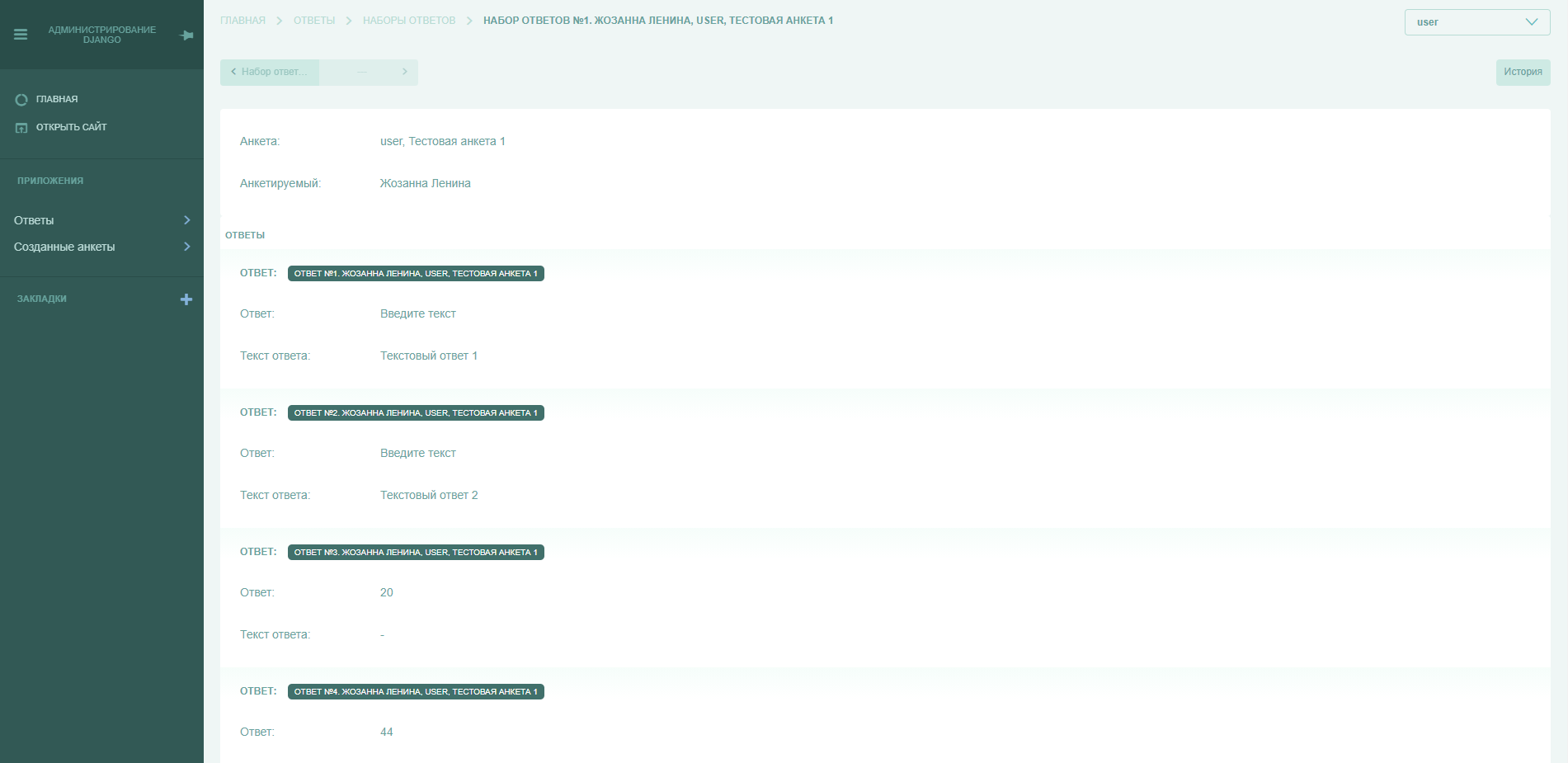
11. Поиск



12. Фильтр по дате



13. Сортировка

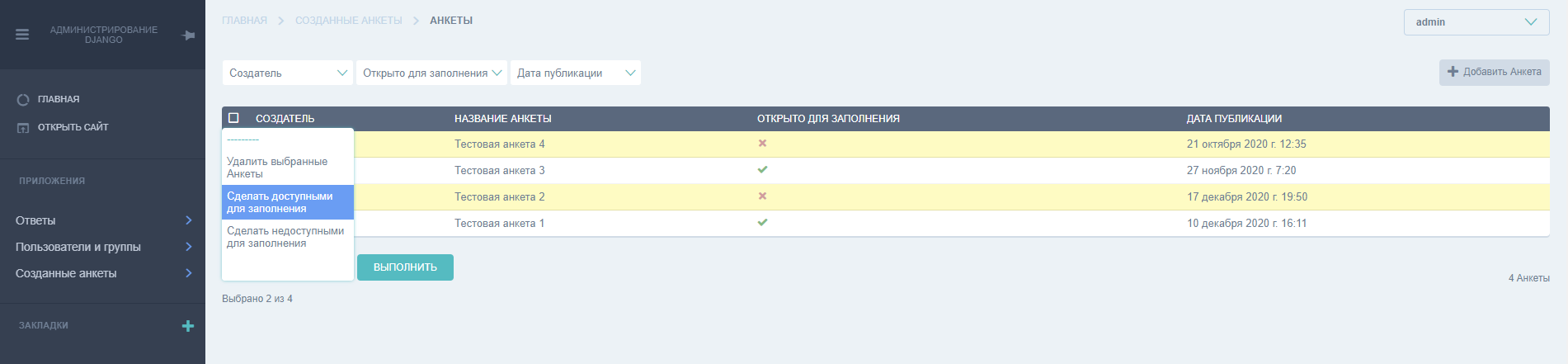


14. Вложенность

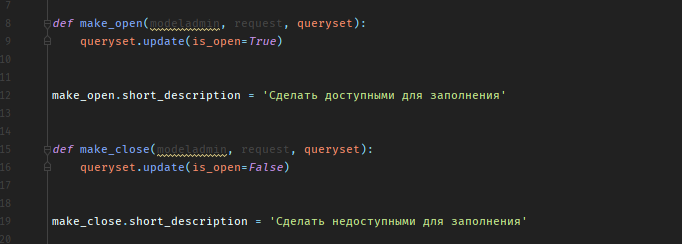


15. Вывод информации из других таблиц

Также в интерфейсе были имплементированы admin-actions. Были реализованы две функции, позволяющие открыть или закрыть для прохождения выбранные анкеты. При необходимости открыть или закрыть несколько анкет сразу, можно использовать admin-actions, а не редактировать каждую анкеты по отдельности.



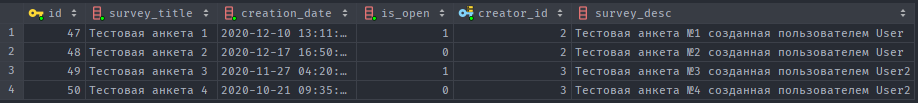
16. Admin-actions



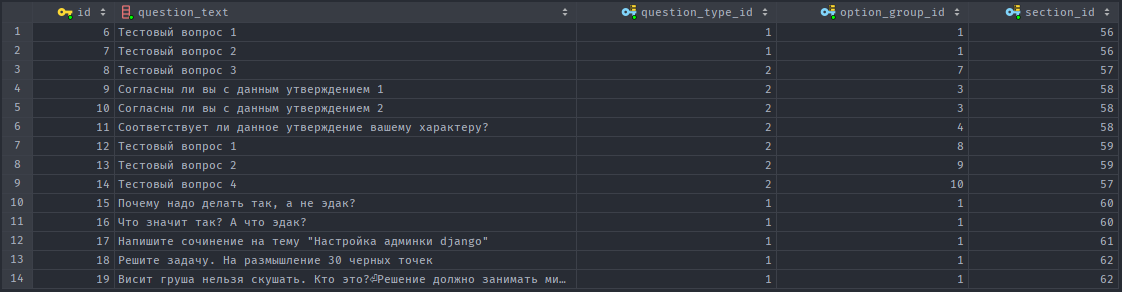
17. Код для admin-actions

Наполнение базы данных.

На данном этапе база данных разрабатываемого сервиса была заполнена тестовыми данными.



18. Таблица анкет



19. Таблица вопросов

Также были разработаны кейсы использования приложения пользователем:

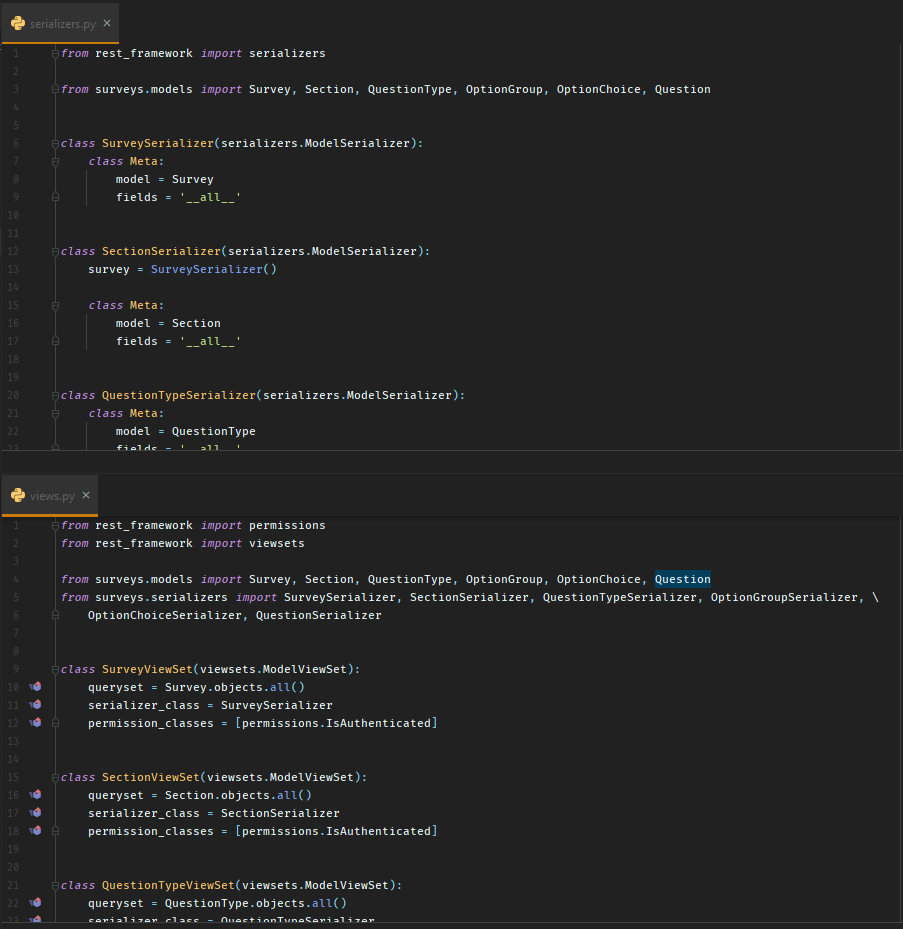
* Редактирование данных пользователя
* Удаление сущностей в приложении анкет
* Удаление сущностей в приложении ответов
* Добавление сущностей в приложении анкет
* Редактирование сущностей в приложении анкет
* Просмотр сущностей в приложении ответов

и администратором:

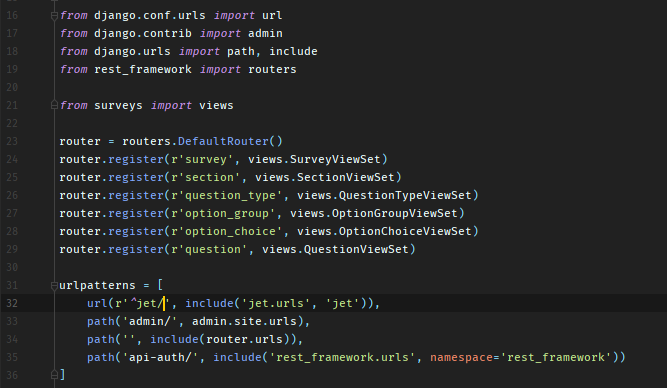
* Редактирование прав пользователя
* Создание групп пользователей с определенными правами
* Редактирование групп пользователей с определенными правами
* Редактирование данных пользователя
* Редактирование списка типов вопросов
* Удаление сущностей в приложении анкет, если они не удовлетворяют требованиям сервиса (неприемлемое содержание и т.п.)
* Удаление сущностей в приложении ответов, если они не удовлетворяют требованиям сервиса (неприемлемое содержание и т.п.)
* Добавление сущностей в приложении анкет
* Добавление сущностей в приложении ответов
* Редактирование сущностей в приложении анкет
* Редактирование сущностей в приложении ответов

Реализация REST API

На данном этапе был установлен Django REST API, создан и настроен serializer.py, добавлены URL.

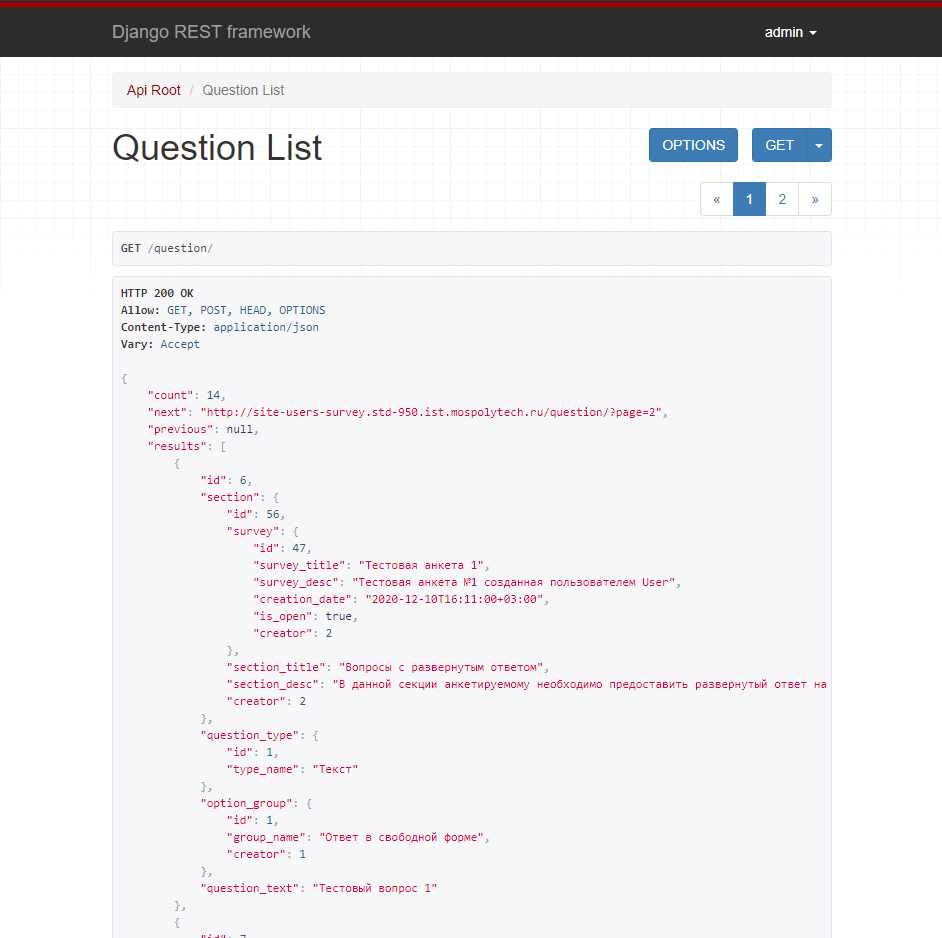


20. serializer.py и views.py



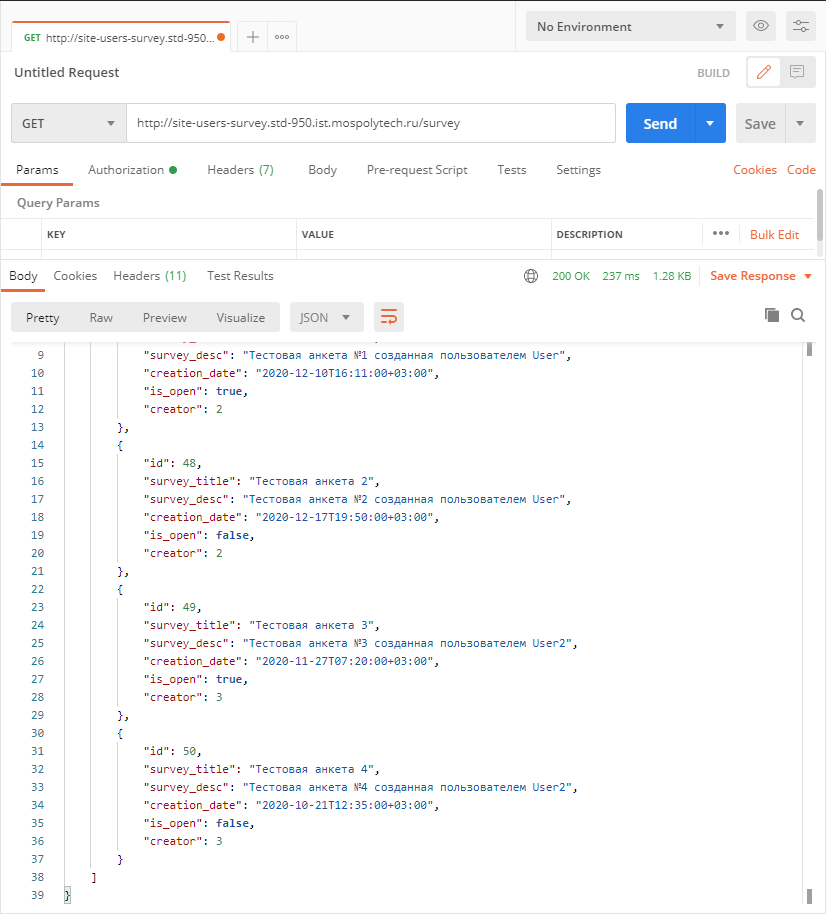
21. urls.py

В итоге получились страницы следующего типа. На них отображается GET запрос по соответствующему serializer-у.

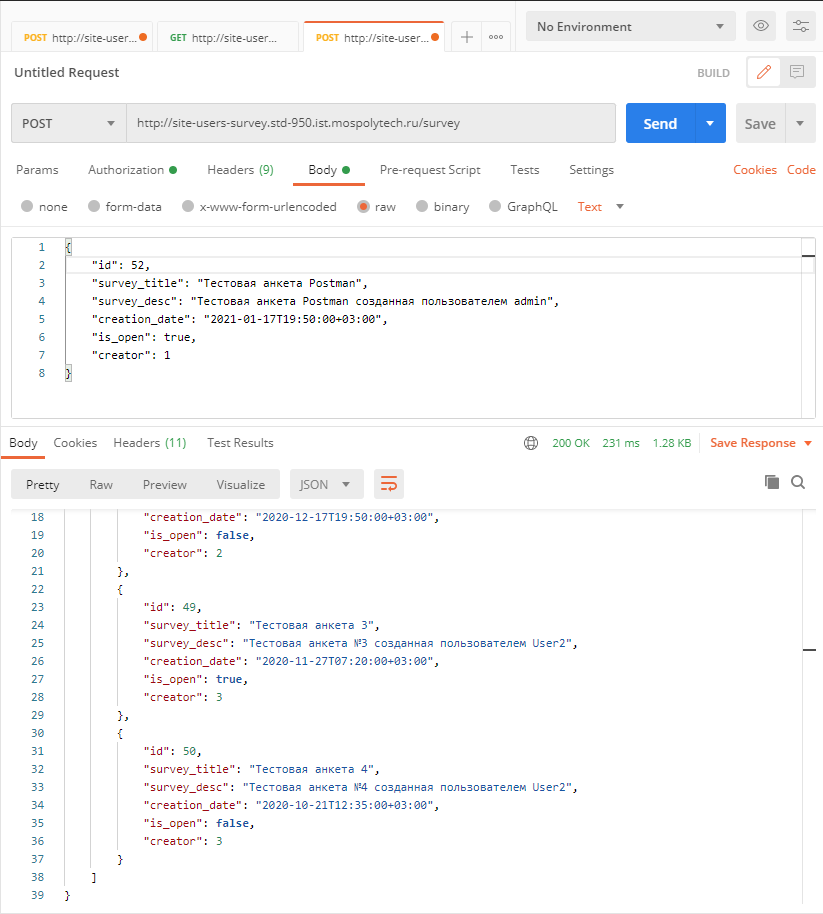


22. Страница REST API

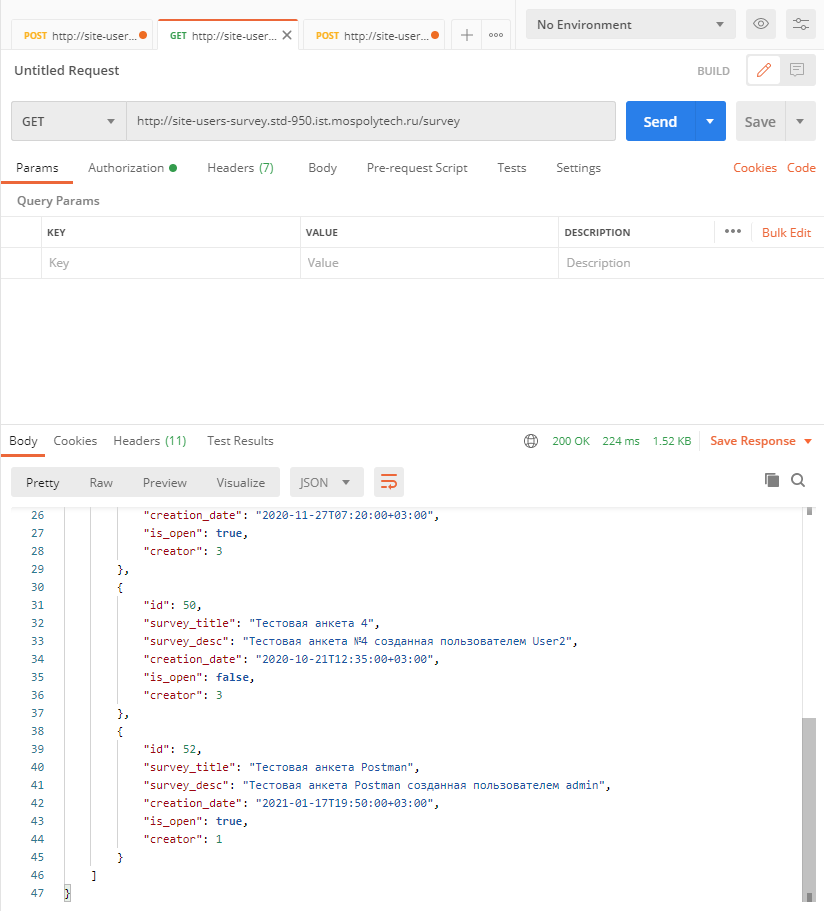
Далее API было протестировано в программе Postman.



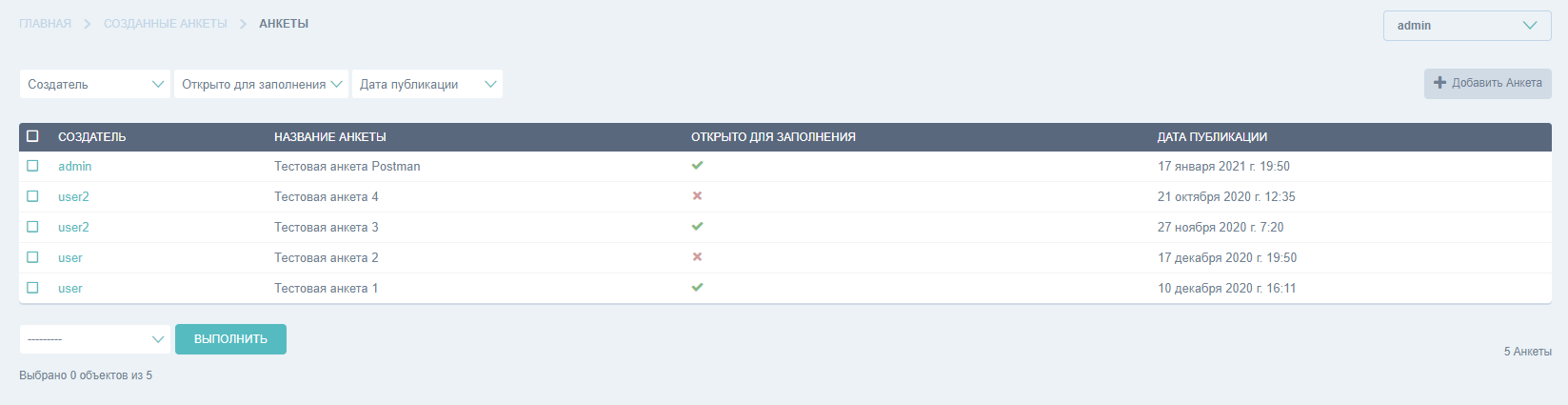
23. Запрос перед добавлением



24. Запрос-добавление



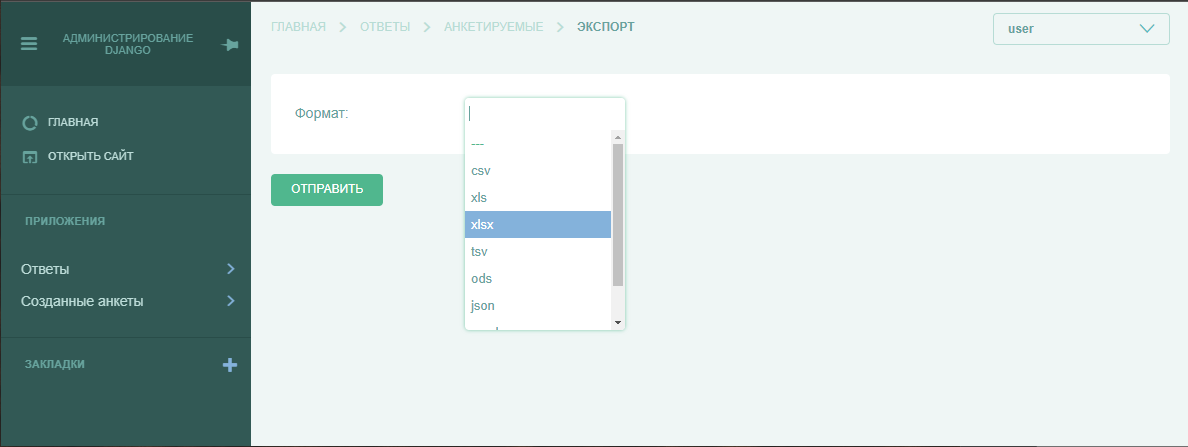
25. Запрос после добавления



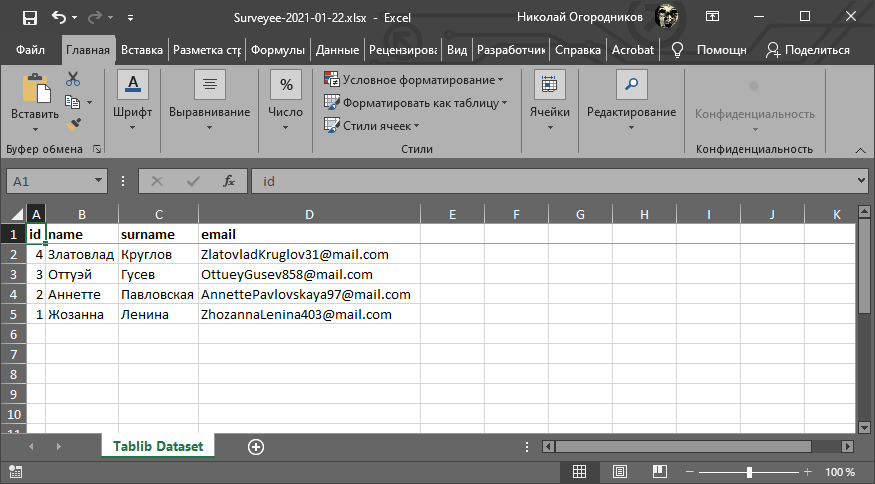
26. Админ-панель после добавления

Реализация импорта и экспорта данных

С помощью django-import-export были реализованы возможности импорта и экспорта данных.



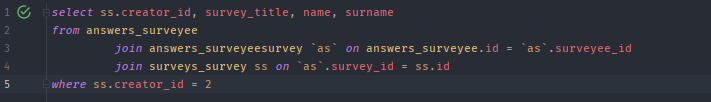
27. Страница экспорта



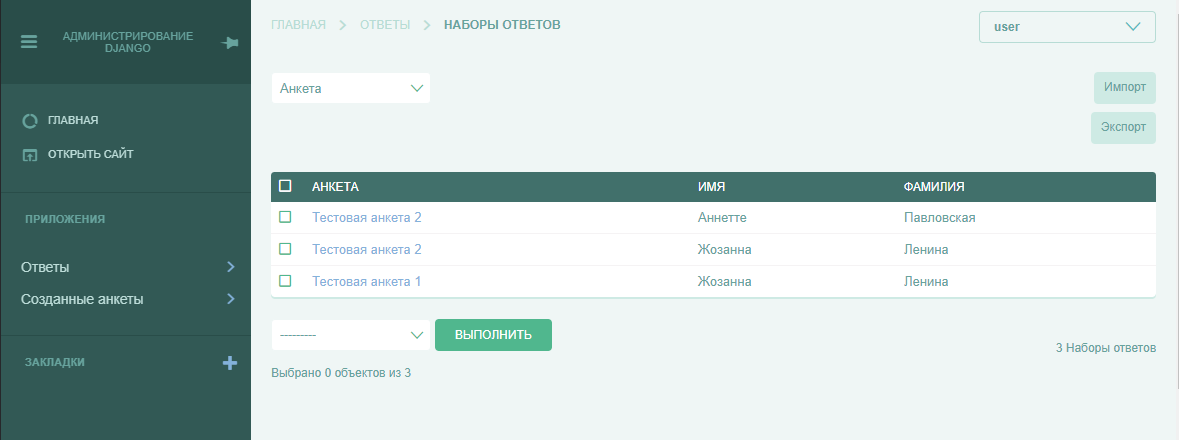
28. Экспортированные данные в формате .xlsx

Типовые запросы к базе данных

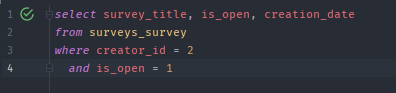
Далее представлены типовые запросы к базе данных и соответствующие им представления в панели администратора.



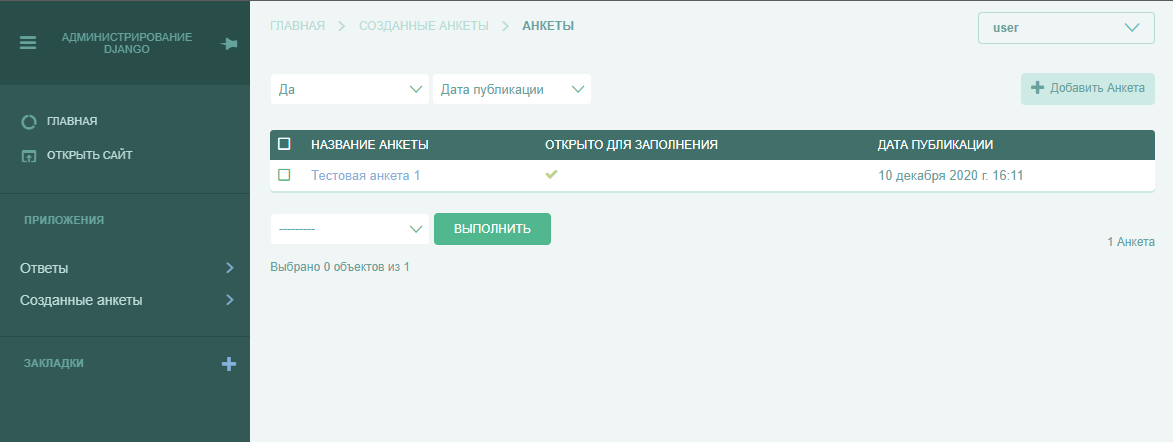
29. Типовой запрос 1



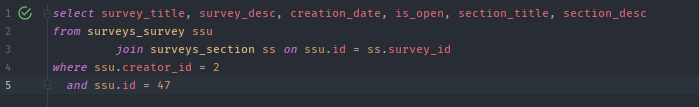
30. Представление типового запроса 1



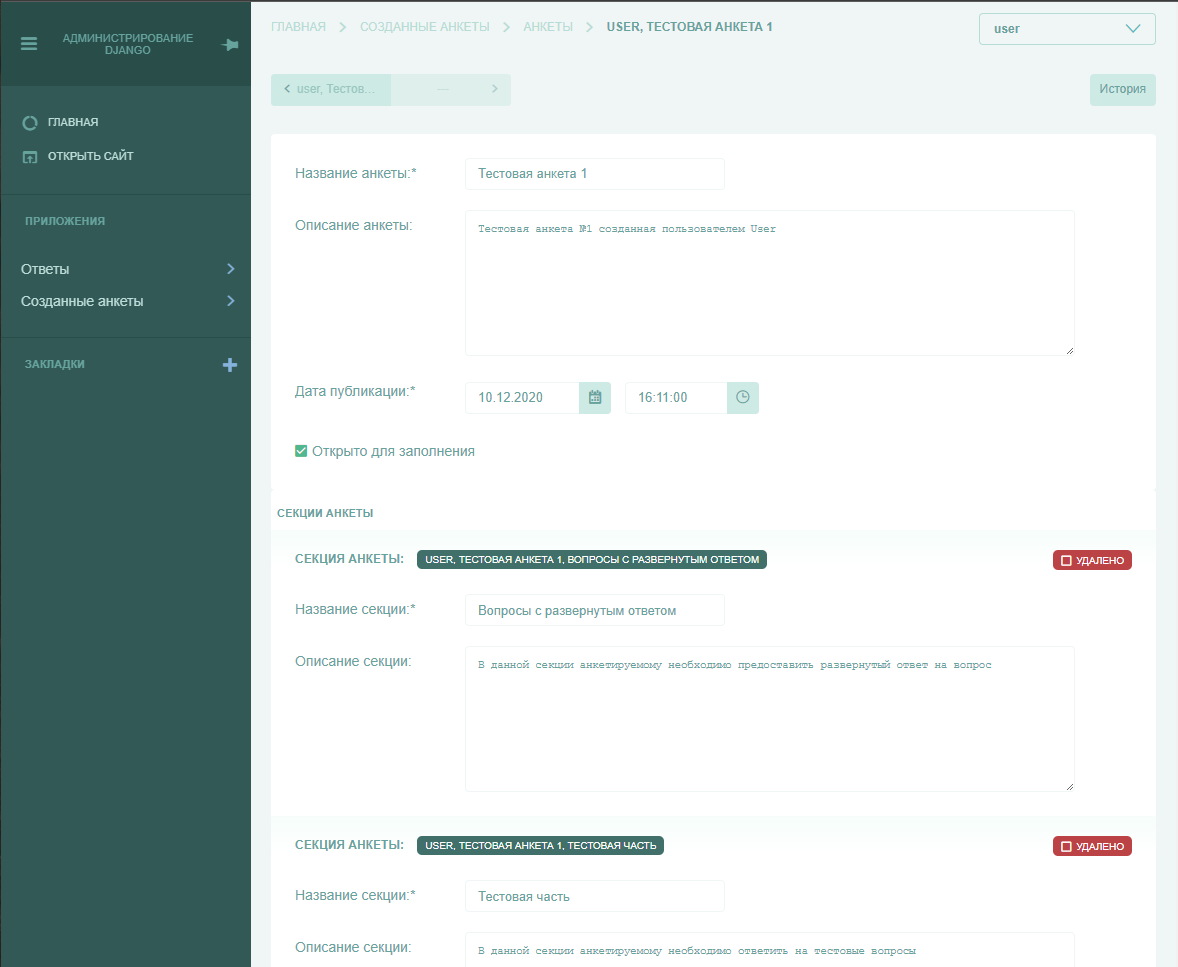
31. Типовой запрос 2



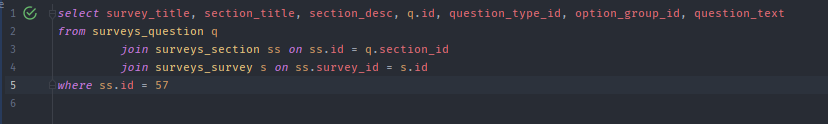
32. Представление типового запроса 2



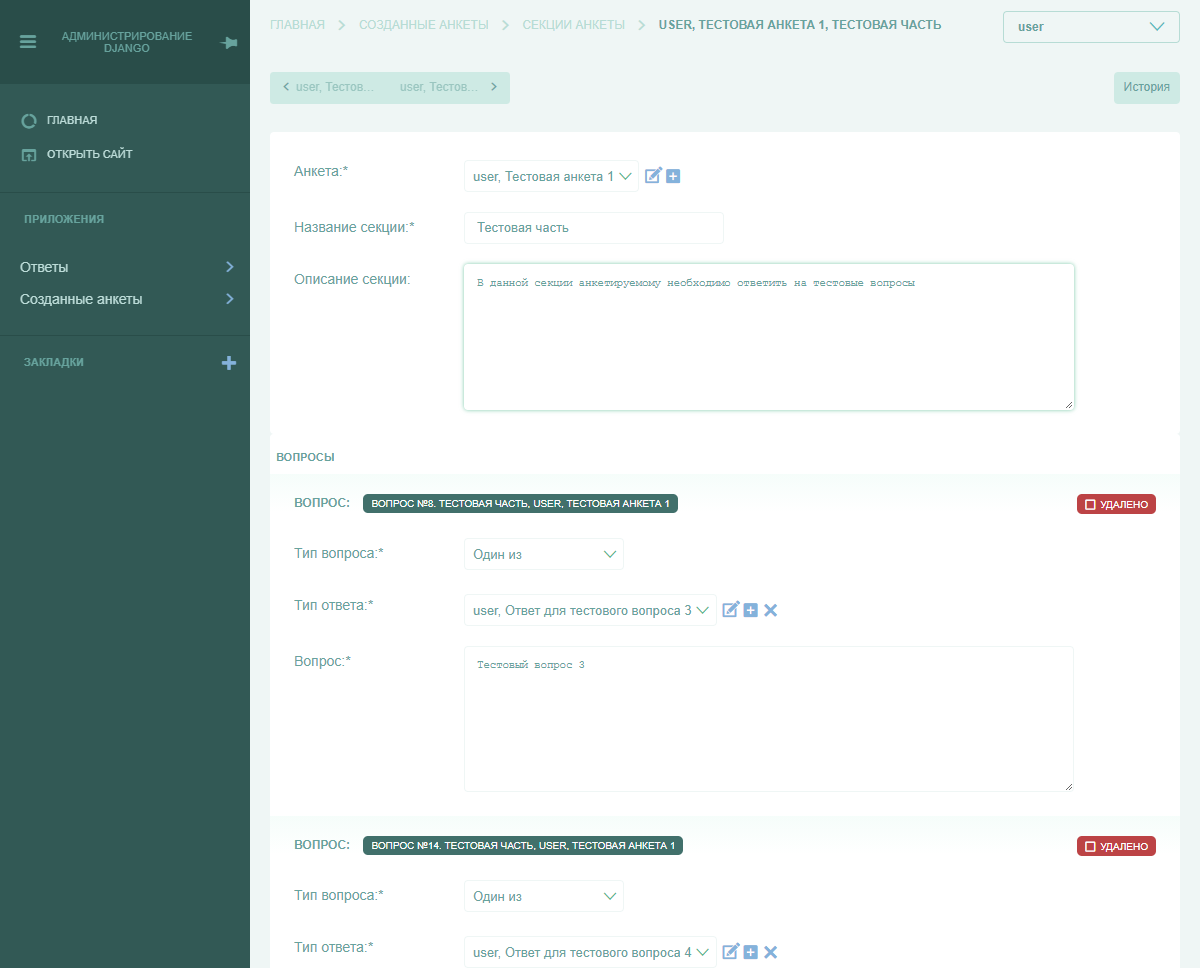
33. Типовой запрос 3



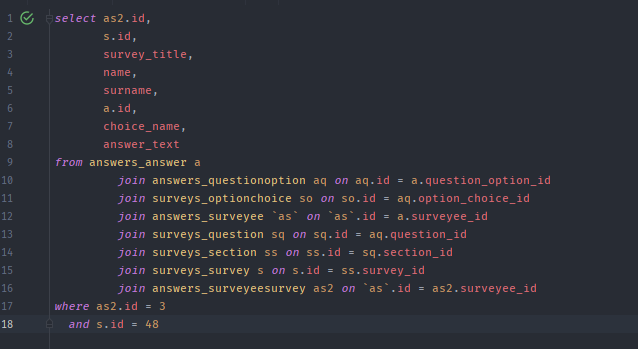
34. Представление типового запроса 3



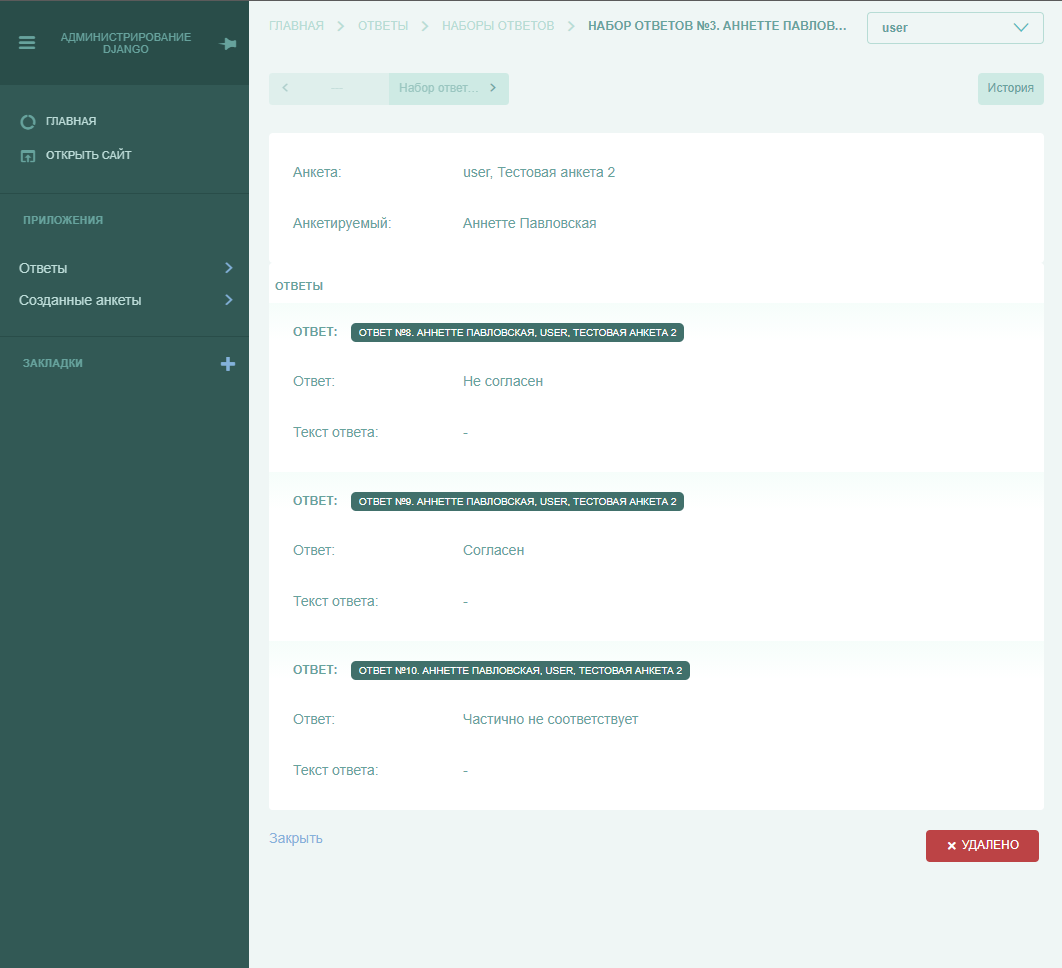
35. Типовой запрос 4



36. Представление типового запроса 4



37. Типовой запрос 5



38. Представление типового запроса 5

Заключение

В результате процесса разработки получилось то, что и изначально планировалось: приложение с настроенной административной панелью на фреймворке Django.

Также в процессе создания данного приложения я применил знания и навыки, полученные за прошлые курсы обучения (python-разработка, проектирование баз данных), освежил их и даже в некоторой степени развил.

Полноценная Full-Stack разработка была новым и интересным опытом в моей практике, и я намерен дальше совершенствовать свои навыки в данном направлении.

Git-репозиторий: <https://github.com/kolaqsq/Site_users_survey>

Приложение: <http://site-users-survey.std-950.ist.mospolytech.ru/admin/>

Список используемых источников

<https://docs.djangoproject.com/en/3.1/>

<https://webdevblog.ru/kak-prevratit-adminku-django-v-legkovesnuju-panel-instrumentov>

<https://djbook.ru/>

<https://django-import-export.readthedocs.io/en/latest/>

<https://www.django-rest-framework.org/>

<https://django.fun/tutorials/put-ot-request-do-response-v-django/>

<https://pythonist.ru/kastomizacziya-admin-paneli-django/>